

# プロセス運転監視システム「プロセスモニタ」

## Process Operating and Monitoring System "Process Monitor"

技術開発本部

広岡隆志

Takashi Hirooka

橋岡啓司

Keiji Hashioka

(化)プラント部 プラント設計課

橋口良徳

Yoshinori Hashiguchi

A multi-purpose process operating and monitoring system by a personal computer, "Process Monitor" has been developed.

The exchange of handlers makes it possible for "Process Monitor" to be applied to any instrument, and its utility programs make it easier to build up the system for a specific process. Flexibility and high productivity of software have been achieved by these features. High-speed and high-performance have also been carried out by a real-time multitask OS.

パソコン用の汎用プロセス運転監視システム「プロセスモニタ」を開発した。

ハンドラを変更することにより、各種の機器に対応が可能で、ユーティリティによって専用のシステムを簡単に構築できるようになっている。そしてこのような特長により、柔軟性、ソフトウェアの生産性に優れ、またリアルタイムマルチタスクOSを使用することにより、高速で高機能なシステムを実現した。

### まえがき

最近、パーソナルコンピュータ(以下、パソコンという)を利用した計測・制御システムがメーカ、ユーザを問わず広く普及している。パソコンは、数年前のイメージと大きく異なり、ある程度の対策を取ることで、産業用に应用することが可能であり、現在の普及はその現れと言える。パソコンを利用するメリットは多くある。まず、経済性に富み、高機能性(カラーグラフィック表示、大容量メモリー、漢字表示、外部記憶装置、各種周辺装置、etc.)を有し、さらに、汎用ソフトウェア(OS, 言語, アプリケーション)が豊富であることである。

当社においても、ここ数年来パソコンを用いた運転監視システムを開発し、納入してきた。しかしながら、パソコンという性格上、BASIC言語が主体となり、高度な機能の実現にはいろいろと問題があった。それは、

- (1) パソコン内蔵のBASIC言語はインタプリタ型であるため処理スピードが遅い。
  - (2) 割り込みに対するサポートが弱いためリアルタイム処理性にかける。
  - (3) シングルタスク型OSであるため多くの機能をプログラムに組み込むには限界がある。
- などである。また一方では、
- (4) システムの多種多様性によりプログラムの作成は一品料理となりソフトウェアの生産性が悪い。

などの問題もあった。

この経験を生かし、このたび、多くのプロセスに汎用的に应用できるプロセス運転監視システム「プロセスモニタ」を開発した。

「プロセスモニタ」は、プロセスの運転状態の監視、デ

ータ収集、画面表示などの機能をパソコンで実現するためのシステムである。必要により、制御機能を付加することも可能である。

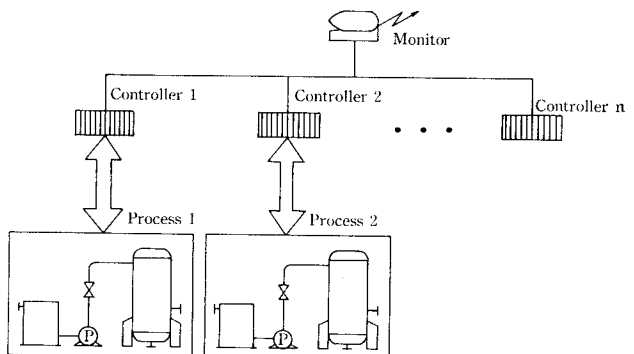
以下「プロセスモニタ」について、概要と特長、システム構成、ソフトウェア機能、応用について説明する。

## 1. 概要と特長

### 1.1 概要

一般に計測制御システムは、第1図に示すようにプロセス、制御機器、監視装置などから構成され、各機器、装置の役割機能分担が決められている。プロセスは、各種水処理プロセス、化学プロセス、または、研究室、試験室での実験装置などである。また制御機器としては、PC (Programmable Controller: プログラマブルコントローラ)、DDC (Direct Digital Controller: デジタル調節計)、データ収集機器などが用いられプロセスのシーケンス制御、アナログ量の調節、データの収集などを行う。さらに上位の監視装置としてパソコンなどのプロセッサを用いることでインテリジェントな機能を付加することができ、運転監視、データロギング、プログラム制御などができる。

この段階でのパソコン利用において、一般に、コントローラの機種、台数、内部点数などの面がシステムごとにより、プログラムは一品料理となっている。これは、プロ



第1図 計測制御システム

Fig. 1 Typical measuring and control system

グラムの生産性、保守性の面からいって、好ましいことではなく問題になっている。

「プロセスモニタ」はこの段階での使用を目的に開発したプログラムレスで使える汎用ソフトウェアであり、制御機器の機種を選ばないという新しい概念で設計されている。各種の制御機器への対応は、ハンドラ（制御機器と直接データをやり取りするプログラム）を機種ごとに用意してシステムに組み込み、内部では統一化されたコマンドにて処理しているため、多くの市販制御機器を同一概念で使用できる。

また、「プロセスモニタ」のOSには、MS-DOSファイルがアクセスできるリアルタイムマルチタスクOSをもちい、各機能単位にモジュール化したプログラムを並行して走行させているので、機能性、操作性が向上している。

言語には、C言語およびアセンブリ言語を用いており、オブジェクトはコンパクトで高速な処理を可能にしている。

一方、「プロセスモニタ」のシステム構築はユーティリティソフトウェア（以下、ユーティリティソフトという）により、プログラムの知識を必要とせず、簡単に行うことができる。このソフトウェアにより設定・登録されたデータは、ディスク上に保管ができ多品種、多目的への対応も可能である。

「プロセスモニタ」の仕様（適用範囲）を第1表に示す。また、「プロセスモニタ」の機能を第2表に示す。

## 1. 2 特長

「プロセスモニタ」の特長をつぎに掲げる。

- (1) パソコン、市販の制御機器を利用したプロセス運転監視システムが経済的に構築できる。
- (2) リアルタイムマルチタスクOSにより高速、高機能を実現している。
- (3) データディスクは、MS-DOSのファイルとして利用できる。
- (4) ユーティリティソフトにより、簡単に短期間でシステムの構築ができる。
- (5) ユーザにて、制御プログラムを作成し、並行して走らすことができる。
- (6) 既設の各種プロセスにも応用できる。

## 2. システム構成

### 2. 1 ハードウェアシステム

ハードウェア構成を第2図に示す。パソコンはNEC製PC-9800シリーズおよびFC-9801(V)が標準であるが、24時間連続運転の場合はFC-9801(V)を推奨している。メモリは最低384 Kバイト必要であるが、ハンドラの数等システム構成により異なる。外部記憶装置としては、フロッピーディスク(2ドライブ以上)、ハードディスクを使用するが、条件のよい環境ならばフロッピーディスクのみのシステムでも使用可能である。しかし、信頼性、スピードの

第1表 「プロセスモニタ」の仕様

Table 1 Specification of "Process Monitor"

#### Specification of hardware

Computer	• 16-bits personal computer
Display	• 640×400 dots Color CRT display
Disk unit	• Floppy disk unit or hard disk unit
Printer	• Kanji printer
Terminal (Option)	• CRT display terminal (alarm, message, guidance display only)
Controllers	• PC, DDC, data logger, extending board, etc (max 16 devices available)

#### Specification of software

Tag	• Max 1024 points
Event	• Max 256
Tag of event	• Max 8 points
Data logging condition	• Event and/or interval
Data logging tag	• Max 60 points/data log
Trend graph	• Max 16 screens
Trend graph pen	• Max 6 pens/screen
Graphic display	• Max 128 screens(depend on disk capacity)
Dynamic display of tag	• Max 64 points/graphic screen
User task	• Max 8 tasks

第2表 「プロセスモニタ」の機能

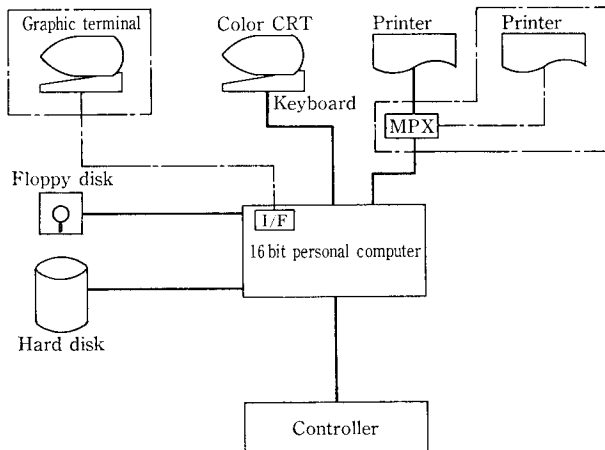
Table 2 Functions of "Process Monitor"

Function	Contents
Event monitor	• Alarm • Message • Status
Data logging	• Interval driven type • Event driven type
Graphic display	• Process flow • Process operation information
Trend graph	• Real time trend graph • Historical trend graph
Alarm display	• Real time alarm display (printer and/or terminal output) • Historical alarm display
Report output	• Daily, monthly report
Control	• Control with special algorithm
System build up	• Utility software package is available

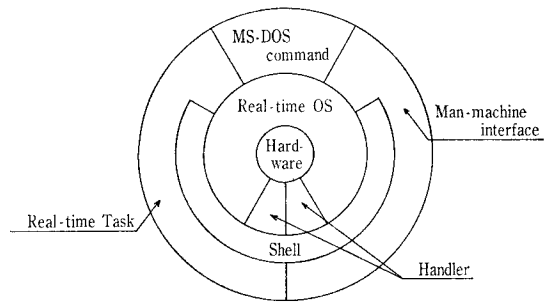
点でハードディスクの使用を推奨する。

CRTは640×400ドットのカラーCRTが必要で、オプションでキャラクタターミナルを接続できる。この場合キャラクタターミナルは、メッセージとガイダンス専用として使用される。

プリンタは漢字プリンタ1台を標準とし、この場合アラームの発生、消滅、日報、月報を1台で印字する。もちろん、適切な優先順位が設定されており、アラームと日報が入り交じるようなことはない。また、マルチプレクサによ



第2図 ハードウェア構成図  
Fig. 2 Hardware configuration



第3図 ソフトウェア構成図  
Fig. 3 Software configuration

り2台のプリンタを切り替えて、アラーム用と日報・月報用に分けて使用することも出来る。

そのほかに、シーケンサ、DDC、データ収集機器等のプロセス制御機器や計測機器とのインターフェイスボードが必要である。現在のところ、RS-232C、GPIB、RS-422、ビットバス等が使用可能で、そのほかの専用通信回線でも、インターフェイスボードがあれば対応可能である。通常はRS-232Cがよく使われるが、この場合同じボーレートであれば、マルチプレクサの使用によりチャンネル数を増加させることもできる。

## 2.2 ソフトウェアシステム

### 2.2.1 ソフトウェア構成

ソフトウェア構成を第3図に示す。リアルタイム・マルチタスクOSと、接続機器ごとのハンドラが、本システムのベースである。その上層にシステムの構成や動作を規定するシステム定数を管理するSDM (System Data Manager)と、プロセスデータを管理するPDM (Process Data Manager)があり、各タスクからの要求に応えるようになってきている。PDMは、プロセスデータの入出力のためにハンドラを管理している。このSDMとPDMを合わせて“シェル”と呼ぶことにする。

システム立ち上げ時にユーティリティで作成されたファイルからSDMに、システム定数(タグコード、イベント等)を展開するシステムイニシャライズタスク、リアルタイムな処理、監視を行うリアルタイムタスク、マンマシンインターフェイスを行うバックグラウンドタスク等が、DOSやシェルの上層にある。

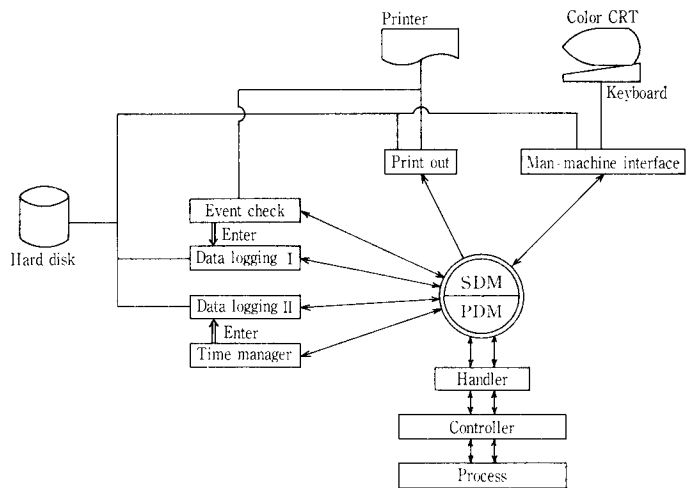
システム定数をファイルに設定するユーティリティは通常オフラインで使用されるが、システム定数によってはオンラインユーティリティも利用できる。

以上が本システムのソフトウェア構成である。図から明らかのように、バックグラウンドタスクとしてMS-DOSのコマンドを実行することも可能である。

以上が本システムのソフトウェア構成である。図から明らかのように、バックグラウンドタスクとしてMS-DOSのコマンドを実行することも可能である。

### 2.2.2 OS

本システムでは、16ビットコンピュータ用OSとしては最も普及しているMS-DOSのファイルを利用できるリ



第4図 ソフトウェア機能図  
Fig. 4 Software function

アルタイムマルチタスクOSを用い、各機能のリアルタイム・マルチタスク処理を可能にしている。ロギングされたデータを何らかの形に処理したいとき、OA用として圧力的に普及したMS-DOS上で作成されたファイルならば、あとの処理が容易である。

以下の説明のために、ソフトウェア機能図を第4図に示す。

### 2.2.3 シェル

#### 1) PDM (プロセスデータ管理)

PDMは、他のタスクからメールボックスによりリード/ライトの要求を受け取り、それを適切なハンドラに手渡す。ハンドラは受け取った要求に従ってリード/ライトのためのコマンドを作成し、プロトコルに従って通信を行い、その結果をPDMに返す。PDMは要求されたタスクに対して結果を返す。内部でやり取りされるデータの表現は、ビットデータは1バイト(0/not 0)、アナログデータは8087の4バイト浮動小数点数(IEEE規格)である。

#### 2) SDM (システムデータ管理)

SDMは各種のシステム定数のテーブルを内部に持ち、

システム立ち上げ時にシステムビルドアップタスクが、これらのテーブルにシステム定数をセットする。システム動作時には、各タスクがこれらのテーブルを参照しながら各種の処理を行う。

### 3. ソフトウェア機能

#### 3.1 リアルタイムタスク

##### 1) 時間管理タスク

このタスクの役割の1つは、時刻をチェックして決められた時刻になると、インターバル起動されるデータロギングタスクを起動することである。

もう1つの役割は、入手した新しいプロセスデータの保持時間をカウントすることである。この処理は次のような意味を持つ。

通常、通信によってプロセス機器からデータを得る場合、複数のデータが一度に返ってくる場合が多い。例えば、シーケンサの入力データが一度に8点分返って来たり、DDCのアナログデータにステータスがついてきたりといった場合である。通信に要する時間は、パソコンの処理時間に比較すると大きいので、せっかく得たデータを捨てるのは大きなロスになる可能性がある。そこで、入手したデータはすべてテーブルに保存し、一定時間(1~5 sec)内は機器をアクセスせずに、このテーブルから値を取り出す。このように、データを保存する時間をカウントするわけである。

##### 2) イベントの監視

このタスクの第一の役割は、イベント(アラーム、メッセージ、ステータス)として設定された値を監視し、その発生、消滅をSDMのイベントヒストリテーブルに記録する。同時に画面の左上に、その発生をオペレータに通知するためにアラーム、メッセージ、ステータスの3つに分けて表示する。また、プリンタが接続されている場合には発生、消滅をプリンタに出力する。

第二の役割としては、データロギングの起動条件として設定されたイベントを監視して条件が満足されれば、データログを起動する。

##### 3) データログ I (インターバル起動)

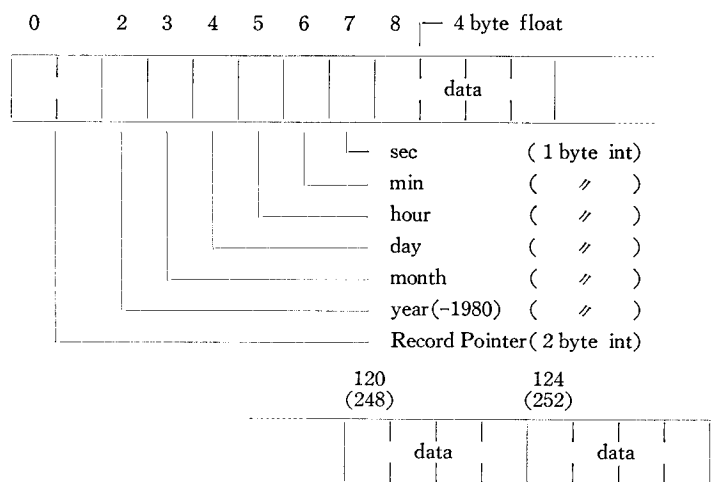
データログのうち、起動条件がインターバルであるもので、起動は前述した時間管理タスクがかかる。

データのフォーマットを第3表に示す。データが30個以下のときは128バイト/レコード、31~60個のときは256バイト/レコードである。

データログ登録の際に「バッチ」と登録したときは、データはロギングファイルの後ろから継ぎ足されて行く。「ロータリ」と登録したときはファイルサイズ256 K バイトを最大として、それ以上になると1レコード目に戻って重ね書きする。これだけでは最終書き込み位置が分からないので、レコードポインタが書き込まれる。レコードポインタは決まった位置にある必要があるため、常に1レコード目に書き込まれ、その他のレコードでは意味を持たない。連続プロセスの場合で、データを残す必要がなければ、「ロータリ」と登録しておけば、常に256 K バイト分だけロギン

第3表 ロギングファイル・フォーマット

Table 3 Logging file format



グファイルが残っていることになる。

データの形式は、レコードポインタが2バイト整数、日時は1バイト整数、データは4バイト浮動小数点数(I E E規格)である。

##### 4) データログ II (イベント起動)

データログのうち、イベントが設定された条件になったときに起動されるもので、ロギングファイルの内容は「データログ I」と同様である。

##### 5) プリンタ出力

日報、月報など登録されたフォーマットのプリント出力を行う。設定時刻毎、またはファンクションキーを押すことにより起動されるが、マルチタスクOSを採用しているので、この間マンマシンインターフェイスは、中断されることなく動作する。

### 3.2 マンマシンインターフェイス

マンマシンインターフェイスはファンクションキーによって切り替えることができ、同じタスクで複数のデータを扱えるもの(トレンド表示、グラフィック表示など)は、さらにデータの切り替えもファンクションキーで行う。以下に各タスクについて説明する。

#### 3.2.1 グラフィック表示

このタスクは、プロセスフローや運転状態監視画面の表示に使用され、監視用に適している。

起動され、データが選択されると、まず画面データが読み込まれる。これは、グラフィック画面のうち静止している部分で、ユーティリティソフトによって描かれたものである。つぎにダイナミック表示部がPDMからデータを得てリアルタイムに表示を繰り返す。ダイナミック表示は例えば、バルブの開閉、機器のON/OFF状態を色の変化で、また液面計、温度計などのアナログ計測値をバーグラフ、数値で表示することができる。

#### 3.2.2 リアルタイムトレンド表示

SDMに登録されたスパンで最大6点のデータをリアルタイムにトレンド表示する。スパンの変更や、ファンクションキーによる画面の切り替えが可能である。

### 3. 2. 3 ヒストリカルトレンド表示

ヒストリカルトレンドはロギングファイルからデータを読み出してトレンドグラフとして表示するタスクで、1画面当たり最大6ペンにデータを設定でき表示できる。データログの項で述べたようにロギングデータには「バッチ」と「ロータリ」の2種類があり、それによって表示の仕方が次のように異なる。

#### 1) バッチ

通常は、スタート時から現在までの全てのデータを、時間軸を自動的に決めて表示する。ユーザが時間軸のスパンを変更したいときは、オンラインでSDMに設定可能であり、その場合は最新のデータをスパンの分だけ表示する。もちろん時間軸は過去にさかのぼることができる。

#### 2) ロータリ

ロギングファイルからレコードポインタを読み出して、最終のレコード位置を調べ、最新のデータをスパンの分だけ表示する。この場合も、任意のスパンで、任意の時点のデータを表示することができる。

### 3. 2. 4 メッセージ表示とガイダンス表示

イベント管理タスクによりアラームの発生、メッセージの通知、ステータスの変化がオペレータに通知されるが、このタスクはその内容とそれに対するガイダンスを表示する。

イベントヒストリテーブルに記録されているアラーム、メッセージ、ステータスを表示し、それらについてガイダンスがあれば、キー操作によりガイダンスファイルから読み込んで表示する。オペレータはアラーム等を表示して確認キーを押すことで、イベントヒストリテーブルは更新される。

### 3. 3 ユーティリティソフト

ユーティリティソフトによる、ユーザのシステム構築手順を第5図に示す。以下、手順にそって説明する。

#### 1) ディスクドライブの設定

ユーティリティソフトのシステムディスクと構築するデータディスクのドライブ番号の設定を行う。

#### 2) 制御機器の登録

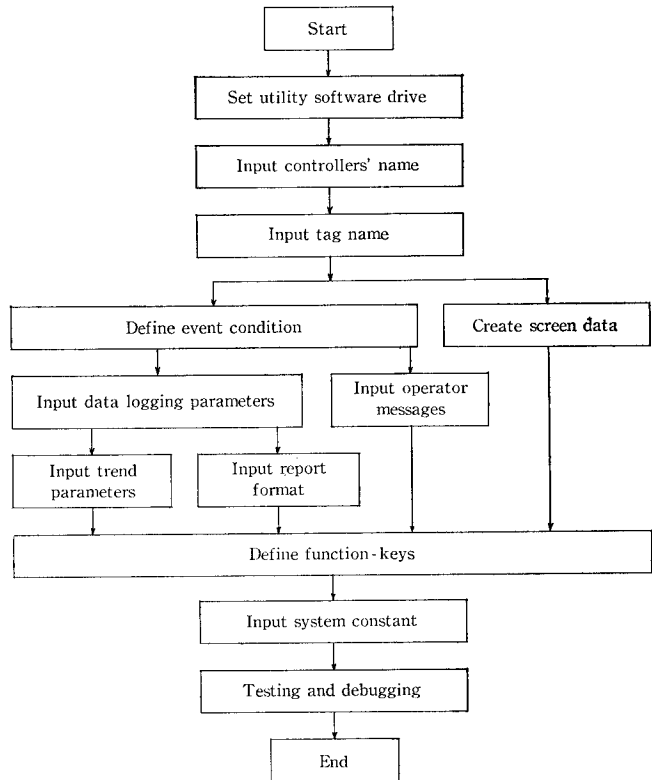
まず、ユーザは適用するプロセスに使用する制御機器を登録する。登録できる制御機器のメーカーと機種には、第4表のものを用意、または、準備しているが、他の機種についても、ハンドラを作成することにより、追加することができる。

#### 3) タグ名の設定

ここで、登録された制御機器は、内部でタグリストに自動展開されるので、ユーザは、このリストにタグ名を設定するだけで各デジタル、アナログの入出力点を参照できるようになる。

#### 4) イベントの登録

つぎに、イベント(事象)を登録する。イベントとは、プロセスのデジタル量、アナログ量の変化を監視し、事象として通知するためのもので、イベント番号に対しタグ



第5図 ユーティリティソフトによるシステム構築手順  
Fig. 5 Flow of building up systems with utility software

第4表 制御機器のメーカーと機種  
Table 4 Available controllers list

Controllers	Maker	Model
PC (Programmable logic controller) or DDC (Direct digital controller)	Izumi Corp.	FA-1 series
	Tateisi Electronics co.	C500 series
	Texas Instruments	PM-550 series
Data transfer device	Mitsubishi Electric co.	K2N series
	Chino Works, Ltd.	SG(X) series
Extend board	Teiryō Incorporated	Tesmic IF series
	Contec	PC module

名を設定する。必要により、各タグ間で、AND(論理積)、OR(論理和)、NOT(否定)の演算条件を付加できる。イベントで登録される各事象の種類としては、つぎの6種があり、ユーザが自由に設定できる。

- AL: アラーム通知(軽故障)
- AM: アラーム通知(中故障)
- AH: アラーム通知(重故障)
- MS: オペレータメッセージ通知
- ST: ステータス変化
- DL: データロギング起動

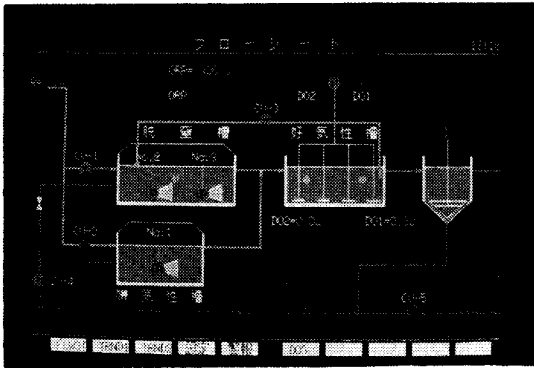


写真1 「リフォーナイトシステム」のプロセスフロー  
Photo.1 Process flow of "Rephonite system"

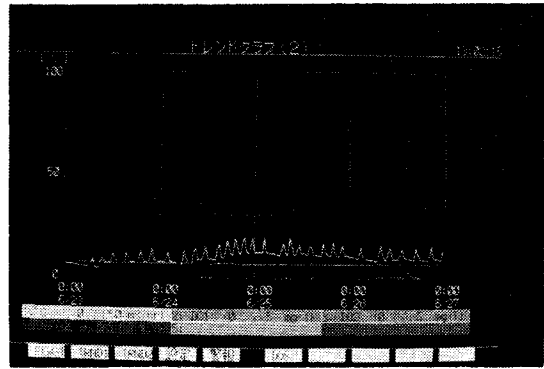


写真2 「リフォーナイトシステム」のヒストリカルトレンド  
Photo.2 Historical trend graph of "Rephonite system"

#### 5) データロギングパラメータの設定

データロギングにはインターバル（時間間隔）起動型と、イベント（事象）起動型の2つのタイプがあり、同時に利用することも可能である。インターバル起動型は、データファイル名およびロギング開始日時と時間間隔を、また、イベント起動型は、データファイル名を条件として入力し、データログするタグ名を設定する。

#### 6) トレンドパラメータの設定

トレンドには、リアルタイム（実時間）トレンドと、ヒストリカル（経過時間）トレンドで、表示するタグ名を画面上の各ペンに設定する。また、同時に、キー表示ラベル、画面タイトルも設定しておく。

#### 7) レポートフォーマットの設定

日報、月報などのレポートのフォーマットを設定する。各項目は、平均、合計などの演算が可能である。

#### 8) メッセージの登録

前述のイベントで登録したアラーム（AL, AM, AH）、オペレータメッセージ（MS）、ステータス（ST）に対応して画面に表示する文字列を登録する。さらに詳細な情報をガイダンスとして登録することができる。

#### 9) グラフィック画面の作成

グラフィック画面は、640×400ドット構成のカラーグラフィック表示ができ、画面上にタグ名を登録することにより、動的な表示が可能である。マウスを利用して、操作性を向上している。

#### 10) キーの定義

キー（ファンクションキー）に各機能モジュールを設定する。

#### 11) システム定数の設定

システムディスクとデータディスクのドライブ番号の設定、アラームのプリンター出力スイッチの設定など、システム定数を設定する。

#### 12) テストおよび修正

以上で、ユーティリティソフトによるシステム構築ができたが、必要により、テストおよび修正を繰り返しシステムが完成する。

## 4. 応 用

「プロセスモニタ」を当社の脱窒脱りん処理プロセスである「リフォーナイトシステム」に応用したので簡単に紹介する。

写真1にプロセスフローを、写真2にヒストリカルトレンドの一例を示す。「リフォーナイトシステム」は、生物学的同時脱窒脱りんプロセスで、実スケールのパイロット装置が現在兵庫県三木市内の污水処理場で運転中である。「リフォーナイトシステム」は、脱窒槽、嫌気槽および硝化槽からなる循環式生物処理プロセスであるが、流入原水の負荷変動（流入量、水質の変動）があると処理水質が悪化する。そこで、流入原水の負荷量を検知し、循環液量、原水分注量（脱窒槽、嫌気槽への原水分注量）のフィードフォワード制御を行っている。そのほか、DO、ORPなどの計測制御、返送汚泥量、引抜汚泥量のプログラム制御なども行っている。また、「プロセスモニタ」を応用したことで、オペレータの操作性の向上、省力化が計られている。

現在、固定床式嫌気性廃水処理装置（ABCシステム）、超純水装置などにも応用している。これらの応用においても同様に、単なる運転監視システムにとどまらず、プロセスのもつ特有の処理ノウハウを制御アルゴリズムとしてプログラムに組み込み、処理性能の向上に効果を発揮している。

## む す び

プロセス運転監視システム「プロセスモニタ」は、各種プロセスに幅広く応用できる汎用的なシステムであり、経済性、機能性、柔軟性そしてソフトウェアの生産性からみて、すぐれたシステムである。今後とも高度化、多様化するユーザのニーズに対応するため、当社独自のシステムを開発して行きたい。

MS-DOSは米国マイクロソフト社の登録商標